

VJEŽBE UTORAK

Kreiranje objekata baze podataka**ZADATAK 1.1**

Kreirajte bazu podataka *Upisi* i tablicu *Polaznici*.



	sifra_polaznika
	ime
	prezime
	mjesto_stanovanja

Rješenje:

```
CREATE DATABASE Upisi;
USE Upisi;
CREATE TABLE Polaznici(
    sifra_polaznika INT,
    ime VARCHAR(50),
    prezime VARCHAR(50),
    mjesto_stanovanja CHAR(25)
);
```

ZADATAK 1.2

U tablicu *Polaznici* dodajte stupac - *broj_telefona* kao cijeli broj.

Rješenje:

```
USE Upisi;
ALTER TABLE Polaznici ADD broj_telefona INT;
```

ZADATAK 1.3

Možda nije najbolje broj telefona pamtiiti kao cijeli broj, zgodnije ga je upamtiti kao tekst jer sadrži crtice, a možda i zagrade, pa promijenite tip stupca u tekstualni.

Rješenje:

```
USE Upisi;
ALTER TABLE Polaznici
    CHANGE broj_telefona broj_telefona VARCHAR(15);
```

ZADATAK 1.4

Izbrišite stupac *broj_telefona* iz tablice *Polaznici*.

Rješenje:

```
USE Upisi;  
ALTER TABLE Polaznici DROP broj_telefona;
```

ZADATAK 1.5

Izbrišite cijelu tablicu *Polaznici*.

Rješenje:

```
USE Upisi;  
DROP TABLE Polaznici;
```

ZADATAK 1.6

Izbrišite bazu podataka *Upisi*.

Rješenje:

```
USE mysql;  
DROP DATABASE Upisi;
```

ZADATAK 1.7

Kreirajte opet bazu podataka *Upisi* i tablicu *Polaznici*. Sada nam stupci *sifra*, *ime* i *prezime* imaju zabranu unosa NULL vrijednosti, tj. prilikom dodavanja novog retka u tablicu obavezno moramo unijeti sva tri polja. *Mjesto_stanovanja* može sadržavati NULL vrijednost. *Šifra* je primarni ključ.

Rješenje:

```
CREATE DATABASE Upisi;  
USE Upisi;  
CREATE TABLE Polaznici(  
    sifra_polaznika INT NOT NULL,  
    ime VARCHAR(50) NOT NULL,  
    prezime VARCHAR(50) NOT NULL,  
    mjesto_stanovanja CHAR(25),  
    PRIMARY KEY (sifra_polaznika)  
);
```

ZADATAK 1.8

Kreirajte preostale dvije tablice *Tečajevi* i *Upisi*. Kreirajte primarne i strane ključeve. Mjesto stanovanja može biti samo Zagreb, Osijek ili Split.

**Rješenje:**

```

USE Upisi;

ALTER TABLE Polaznici ENGINE=InnoDB;

ALTER TABLE Polaznici
  CHANGE mjesto_stanovanja
    mjesto_stanovanja ENUM('Zagreb','Osijek','Split')
    NOT NULL;

CREATE TABLE Tecajevi(
  sifra_tecaja CHAR(3) NOT NULL,
  naziv_tecaja VARCHAR(50) NOT NULL,
  PRIMARY KEY (sifra_tecaja)
) ENGINE=InnoDB;

CREATE TABLE Upisi(
  sifra_polaznika INT,
  sifra_tecaja CHAR(3),
  PRIMARY KEY (sifra_polaznika,sifra_tecaja),
  KEY (sifra_polaznika),
  KEY (sifra_tecaja)
) ENGINE=InnoDB;

ALTER TABLE Upisi
  ADD CONSTRAINT Upisi_ibfk_1
    FOREIGN KEY (sifra_polaznika)
    REFERENCES Polaznici (sifra_polaznika),
  ADD CONSTRAINT Upisi_ibfk_2
    FOREIGN KEY (sifra_tecaja)
    REFERENCES Tecajevi(sifra_tecaja);
  
```


VJEŽBE – SRIJEDA

Ubacivanje podataka u tablicu

ZADATAK 2.1

Popunite tablice podacima kako je prikazano na slici:

Polaznici			
Šifra polaznika	Ime polaznika	Prezime polaznika	Mjesto stanovanja
1	Ana	Milić	Zagreb
2	Sanja	Tarak	Split
3	Mladen	Gork	Osijek
4	Ivana	Matkić	Split
5	Marina	Anić	Osijek
6	Ivica	Limac	Split

Tečajevi	
Šifra tečaja	Naziv tečaja
P01	Osnove rada PC računala
P02	Microsoft Word
N01	SQL - osnove
O01	Računalni operator – uredsko poslovanje
O02	Specijalist poslovne primjene računala
O03	Grafički dizajner

Upisi	
Šifra polaznika	Šifra tečaja
1	P01
2	P02
3	N01
4	O01
5	O02
6	P01

Rješenje:

USE Upisi;

```

INSERT INTO Polaznici VALUES
  (1, 'Ana', 'Milić', 'Zagreb');
INSERT INTO Polaznici VALUES
  (2, 'Sanja', 'Tarak', 'Split');
INSERT INTO Polaznici VALUES
  (3, 'Mladen', 'Gork', 'Osijek');
INSERT INTO Polaznici VALUES
  (4, 'Ivana', 'Matkić', 'Split');
INSERT INTO Polaznici VALUES
  (5, 'Marina', 'Anić', 'Osijek');
INSERT INTO Polaznici VALUES
  (6, 'Ivica', 'Limić', 'Split');

INSERT INTO Tecajevi VALUES
  ('P01', 'Osnove rada PC računala');
INSERT INTO Tecajevi VALUES
  ('P02', 'Microsoft Word');
INSERT INTO Tecajevi VALUES
  ('N01', 'SQL - osnove');
INSERT INTO Tecajevi VALUES
  ('O01', 'Računalni operator - uredsko poslovanje');

```

```
INSERT INTO Tecajevi VALUES  
('002','Specijalist poslovne primjene računala');  
INSERT INTO Tecajevi VALUES  
('003','Grafički dizajner');  
  
INSERT INTO Upisi VALUES (1,'P01');  
INSERT INTO Upisi VALUES (2,'P02');  
INSERT INTO Upisi VALUES (3,'N01');  
INSERT INTO Upisi VALUES (4,'O01');  
INSERT INTO Upisi VALUES (5,'O02');  
INSERT INTO Upisi VALUES (6,'P01');
```

Prikaz podataka iz tablice

ZADATAK 2.2

Ispišite podatke iz svih stupaca tablice *Polaznici*.

Rješenje:

```
USE Upisi;  
SELECT * FROM Polaznici;
```

ZADATAK 2.3

Ispišite podatke iz svih stupaca tablice *Polaznici* i stupcima promijenite naziv u ispisu rezultata: sifra_polaznika – Šifra, ime – Ime osobe, prezime – Prezime osobe, mjesto Stanovanja – Mjesto.

Rješenje:

```
USE Upisi;  
SELECT sifra_polaznika AS `Šifra`,  
       ime AS `Ime osobe`,  
       prezime AS `Prezime osobe`,  
       mjesto Stanovanja AS `Mjesto`  
FROM Polaznici;
```

ZADATAK 2.4

Sortirajmo podatke u upitu tako da je *ime* sortirano uzlazno.

Rješenje:

```
USE Upisi;  
SELECT * FROM Polaznici ORDER BY ime;
```

ZADATAK 2.5

Sortirajmo podatke u upitu tako da je *mjesto_stanovanja* sortirano silazno.

Rješenje:

```
USE Upisi;  
  
SELECT * FROM Polaznici  
ORDER BY mesto_stanovanja DESC;
```

ZADATAK 2.6

Ispišite sva mjesta stanovanja. Na popisu se ne smiju pojaviti duplikati.

Rješenje 1:

```
USE Upisi;  
  
SELECT DISTINCT mesto_stanovanja FROM Polaznici;
```

Rješenje 2:

```
USE Upisi;  
  
SELECT mesto_stanovanja  
FROM Polaznici  
GROUP BY mesto_stanovanja  
ORDER BY mesto_stanovanja;
```

ZADATAK 2.7

Ispišite prva 3 zapisa iz tablice *Polaznici*.

Rješenje:

```
USE Upisi;  
  
SELECT * FROM Polaznici LIMIT 0,3;
```

ZADATAK 2.8

Ispišite sve polaznike iz Splita.

Rješenje:

```
USE Upisi;  
  
SELECT * FROM Polaznici  
WHERE mesto_stanovanja = 'Split';
```

ZADATAK 2.9

Ispišite sve polaznike kojima je šifra 1, 3 ili 5.

Rješenje 1:

```
USE Upisi;
```

```
SELECT * FROM Polaznici  
WHERE sifra_polaznika IN (1,3,5);
```

Rješenje 2:

```
USE Upisi;  
  
SELECT * FROM Polaznici  
WHERE sifra_polaznika = 1  
      OR sifra_polaznika = 3  
      OR sifra_polaznika = 5;
```

ZADATAK 2.10

Ispišite iz tablice *Upisi* sve šifre polaznika koji su upisali seminare „Osnove rada PC računala“ (šifra P01) i „Microsoft Word“ (šifra P02).

Rješenje 1:

```
USE Upisi;  
  
SELECT sifra_polaznika FROM Upisi  
WHERE sifra_tecaja IN ('P01', 'P02');
```

Rješenje 2:

```
USE Upisi;  
  
SELECT sifra_polaznika FROM Upisi  
WHERE sifra_tecaja = 'P01'  
      OR sifra_tecaja = 'P02';
```

ZADATAK 2.11

Ispišite iz tablice *Polaznici* imena i prezimena onih polaznika koje ne žive u Zagrebu.

Rješenje 1:

```
USE Upisi;  
  
SELECT ime, prezime FROM Polaznici  
WHERE mjesto_stanovanja != 'Zagreb';
```

Rješenje 2:

```
USE Upisi;  
  
SELECT ime, prezime FROM Polaznici  
WHERE mjesto_stanovanja <> 'Zagreb';
```


ZADATAK 2.12

Ispišite iz tablice *Polaznici* imena i prezimena onih polaznika koje žive u Splitu i šifra im je veća od 2.

Rješenje:

```
USE Upisi;

SELECT ime, prezime FROM Polaznici
WHERE mjesto_stanovanja = 'Split'
      AND sifra_polaznika > 2;
```

ZADATAK 2.13

Ispišite iz tablice *Polaznici* sve polaznike čije prezime počinje slovom *M*.

Rješenje 1:

```
USE Upisi;

SELECT ime, prezime FROM Polaznici
WHERE prezime LIKE 'M%';
```

Rješenje 2:

```
USE Upisi;

SELECT ime, prezime FROM Polaznici
WHERE LEFT(prezime,1) = 'M';
```

ZADATAK 2.14

Ispišite iz tablice *Tecajevi* sve tečajeve koje sadrže riječ *Račun*.

Rješenje:

```
USE Upisi;

SELECT * FROM Tecajevi
WHERE naziv_tecaja LIKE '%Račun%';
```

ZADATAK 2.15

Ispišite iz tablice *Polaznici* sve polaznike čije prezime završava na *ak* ili *rk*.

Rješenje 1:

```
USE Upisi;

SELECT * FROM Polaznici
WHERE prezime LIKE '%ak'
      OR prezime LIKE '%rk';
```

Rješenje 2:

```
USE Upisi;

SELECT * FROM Polaznici
WHERE RIGHT(prezime,2) = 'ak'
      OR RIGHT(prezime,2) = 'rk';
```

ZADATAK 2.16

Ispišite iz tablice *Polaznici* sva mjesta stanovanja koja u svome nazivu ne sadrže slovo s.

Rješenje:

```
USE Upisi;

SELECT * FROM Polaznici
WHERE mjesto_stanovanja NOT LIKE '%s%';
```

Prikaz podataka iz više tablica

Sve sljedeće zadatke riješite koristeći bazu podataka FAKULTET.

ZADATAK 2.17

Ispišite imena i prezimena nastavnika s nazivom mjesta u kojem stanuju.

Rješenje 1:

```
USE fakultet;

SELECT
    n.imeNastavnik,
    n.prezNastavnik,
    m.nazMjesto
FROM
    nastavnik AS n,
    mjesto AS m
WHERE n.pbrStan = m.pbr;
```

Rješenje 2:

```
USE fakultet;

SELECT
    n.imeNastavnik,
    n.prezNastavnik,
    m.nazMjesto
FROM nastavnik AS n
JOIN mjesto AS m
    ON n.pbrStan = m.pbr;
```

ZADATAK 2.18

Ispišimo imena i prezimena svih studenata zajedno s ispitima na koje su izašli i ocjenama koje su dobili. Naravno, bit će studenata koji nisu ni jednom izašli na neki ispit, pa kod njih ne možemo ispisati šifru ispita.

Rješenje:

```
USE fakultet;

SELECT
    s.imeStud,
    s.prezStud,
    i.datIspit,
    i.ocjena,
    p.nazPred
FROM stud AS s
LEFT JOIN ispit AS i ON s.mbrStud = i.mbrStud
LEFT JOIN pred AS p ON i.sifPred = p.sifPred;
```

ZADATAK 2.19

Ispišimo sve predmete i dvorane u kojima se predaje. Kako se neki predmeti ne predaju ovaj semestar, oni nemaju rezerviranu dvoranu, pa će to biti reprezentirano NULL vrijednostima.

Rješenje:

```
USE fakultet;

SELECT p.*, d.*
FROM pred AS p
LEFT JOIN rezervacija AS r ON p.sifPred = r.sifPred
LEFT JOIN dvorana AS d ON r.oznDvorana = d.oznDvorana;
```

ZADATAK 2.20

Ispišimo sve studente koji su izašli na ispite i sve predmete. Naravno, bit će studenata koji nisu izašli ni na jedan predmet, ali i predmeta na koje nitko nije izašao.

Rješenje:

Da bismo si olakšali posao, kreirajmo prvo jedan pogled:

```
USE fakultet;

CREATE VIEW StudentiIspiti AS
    SELECT s.mbrStud, s.imeStud, i.sifPred, i.ocjena FROM ispit i
    RIGHT OUTER JOIN stud s ON i.mbrStud = s.mbrStud;
```

Zatim, spojimo s predmetima:

```
SELECT mbrStud, imeStud, ocjena, p.sifPred, nazPred
    FROM StudentiIspiti s
    RIGHT OUTER JOIN pred p ON s.sifPred = p.sifPred
UNION
SELECT mbrStud, imeStud, ocjena, p.sifPred, nazPred
    FROM StudentiIspiti s
    LEFT OUTER JOIN pred p ON s.sifPred = p.sifPred
ORDER BY mbrStud ASC;
```

ZADATAK 2.21

Ispišite sve organizacijske jedinice s pripadajućim nadređenim organizacijskim jedinicama.

Rješenje:

```
USE fakultet;  
  
SELECT o1.nazOrgjed AS `Organizacijska jedinica`,  
       o2.nazOrgjed AS `Nadređena jedinica`  
FROM orgjed o1  
INNER JOIN orgjed o2 ON o2.sifOrgjed = o1.sifNadorgjed;
```

VJEŽBE – ČETVRTAK

Izmjene podataka u tablicama

Sve sljedeće zadatke riješite koristeći bazu podataka UPISI.

ZADATAK 3.1

Promijenimo prezime polaznice Ane Milić u Morsan.

Rješenje:

```
USE Upisi;  
  
UPDATE Polaznici  
SET prezime = 'Morsan'  
WHERE ime = 'Ana' AND prezime = 'Milić';
```

ZADATAK 3.2

Iz tablice *Upisi* izbrisati podatke o polazniku sa šifrom 6.

Rješenje:

```
USE Upisi;  
  
DELETE FROM Upisi  
WHERE sifra_polaznika = 6;
```

ZADATAK 3.3

Izbrišite sve podatke iz tablice *Upisi*.

Rješenje:

```
USE Upisi;  
  
TRUNCATE TABLE Upisi;
```

ZADATAK 3.4

Obrišite tablicu *Upisi*.

Rješenje:

```
USE Upisi;  
DROP TABLE Upisi;
```

ZADATAK 3.5

Obrišite cijelu bazu podataka *Upisi*.

Rješenje:

```
USE mysql;  
DROP DATABASE Upisi;
```

Sve sljedeće zadatke riješite koristeći bazu podataka FAKULTET.

ZADATAK 3.6

Kreirajte tablicu *Osjecani* u kojoj se nalaze podaci samo studenata iz tablice *stud* čije je mjesto stanovanja Osijek.

Rješenje:

```
USE fakultet;  
CREATE TABLE Osjecani  
  SELECT s.*  
  FROM stud AS s  
  JOIN mjesto AS m  
    ON m.pbr = s.pbrStan  
  WHERE m.nazMjesto = 'Osijek';  
SELECT * FROM Osjecani;  
SHOW TABLES;
```

ZADATAK 3.7

Kreirajte privremenu tablicu *Splicani* u kojoj se nalaze podaci samo studenata iz tablice *stud* čije je mjesto stanovanja Split.

Rješenje:

```
USE fakultet;  
CREATE TEMPORARY TABLE Splicani  
  SELECT s.*  
  FROM stud AS s  
  JOIN mjesto AS m  
    ON m.pbr = s.pbrStan  
  WHERE m.nazMjesto = 'Split';  
SELECT * FROM Splicani;  
SHOW TABLES;
```

ZADATAK 3.8

Kreirajte pogled na tablicu *stud*. Sadrži samo imena i prezimena studenata.

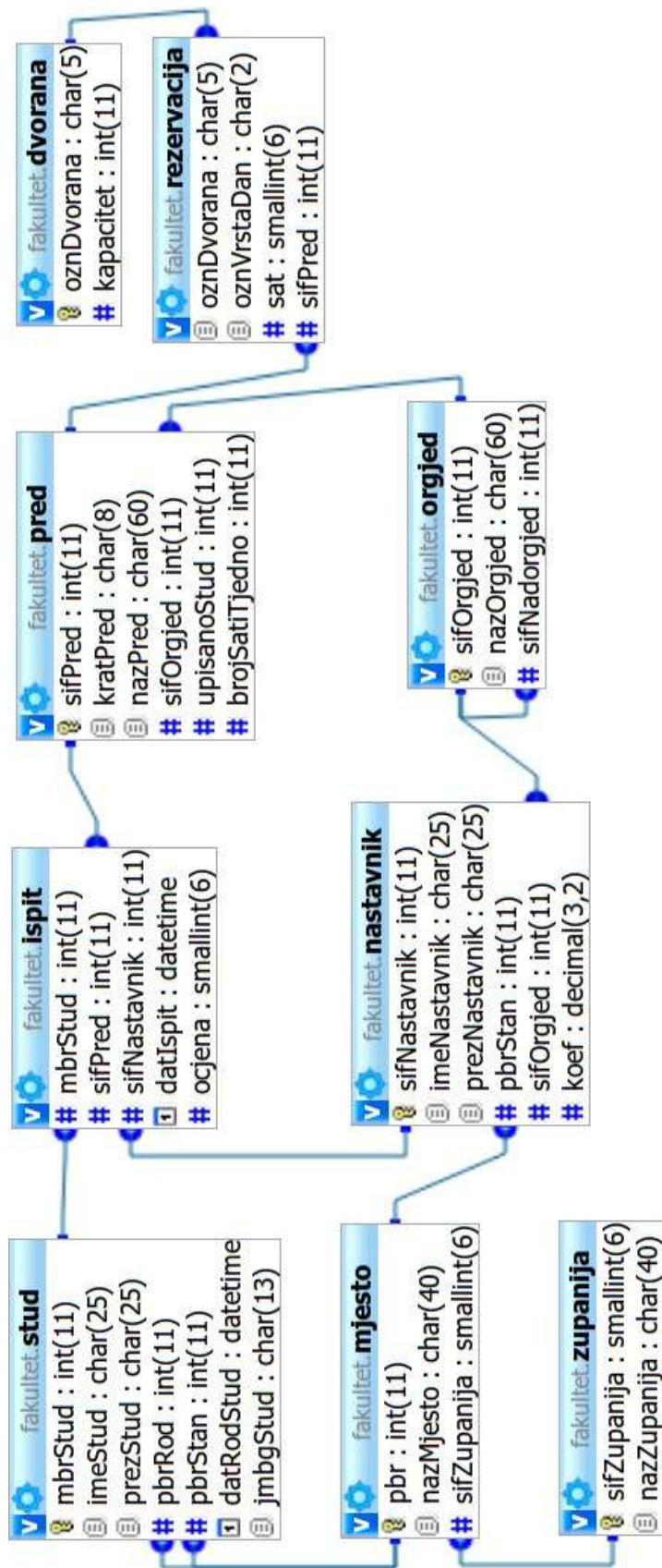
Rješenje:

```
USE fakultet;
```

```
CREATE VIEW Studenti AS  
  SELECT imeStud, prezStud  
  FROM stud;
```

```
SELECT * FROM Studenti;
```

DODATAK A – FAKULTETSKA BAZA



Oznaka znači primarni ključ (engl. *Primary Key*). Linije označavaju relacije među tablicama, pri čemu je sa jedne strane uvijek primarni ključ, a sa druge strani ključ (engl. *Foreign Key*). Na vrhu je naziv baze i tablice u obliku baza.tablica, pri čemu je podebljan naziv tablice. Oznaka označava tekstualno polje. Oznaka označava brojčano polje. Oznaka označava datumsko/vremensko polje.